UNIVERSITY OF GRONINGEN
OPTICAL CONDENSED MATTER PHYSICS

USERGUIDE

Pulse duration measurement software

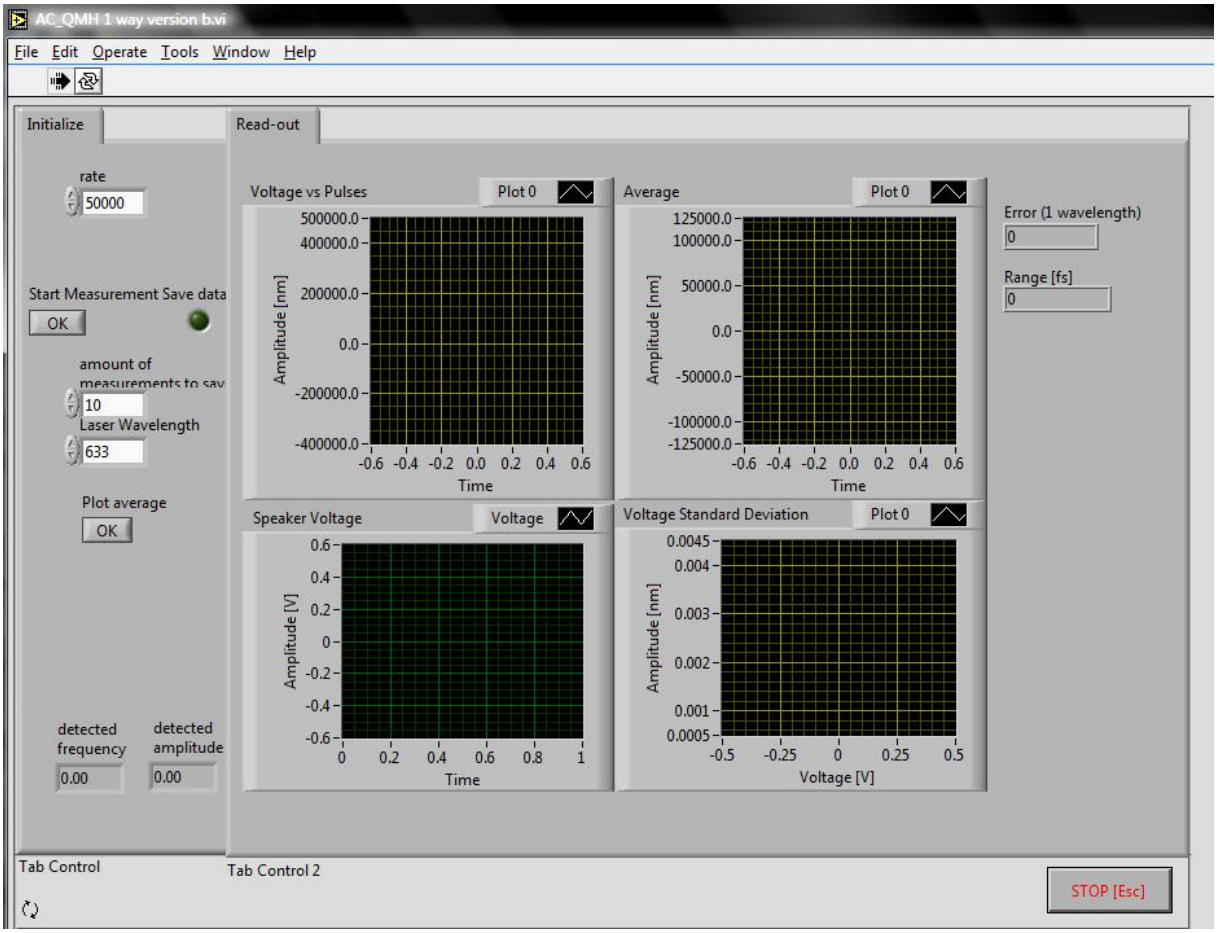
A.J. Oostra

1-10-2009

Userguide for the pulse duration measurement programs: calibration software and the main program.

Calibration software

The calibration software can be used to calibrate the main program. It works by calculating the optical displacement, and placing it in a file against the voltage of the speaker. The optical displacement is calculated by counting the fringes of the interference pattern. The input for the speaker voltage is channel ai3, and the for the LED-voltage ai1.



HeNe.exe / AC_QMH 1 way version b.vi

Rate: rate at which the DAQ-card is read.
Save data: click on button to have the data be stored when running the program.
Amount of measurements to save: self-explanatory
Laser wavelength: self-explanatory
Plot average: Loads n (same as amount saved) files and combines them before plotting and storing the averaged results.
Voltage vs pulses: Optical displacement vs speaker-voltage
Average: Graph of the averaged optical displacement vs. speaker-voltage
Speaker voltage: The speaker voltage vs time
Stop: self-explanatory

Run the program by clicking on the “start measurement” button. If the light of saving is not on, no files will be stored. In case it’s turned on, it will store the defined amount of measurements. This will be done by asking once for a name to store the data to. After asking it will store each measurement in a file with increasing file names (…0,…1,…n).

Click on the average button to load n files (same as amount stored) and have it be plotted in the average-graph. The program then stores the calibration file.

Software’s main-subVI’s:

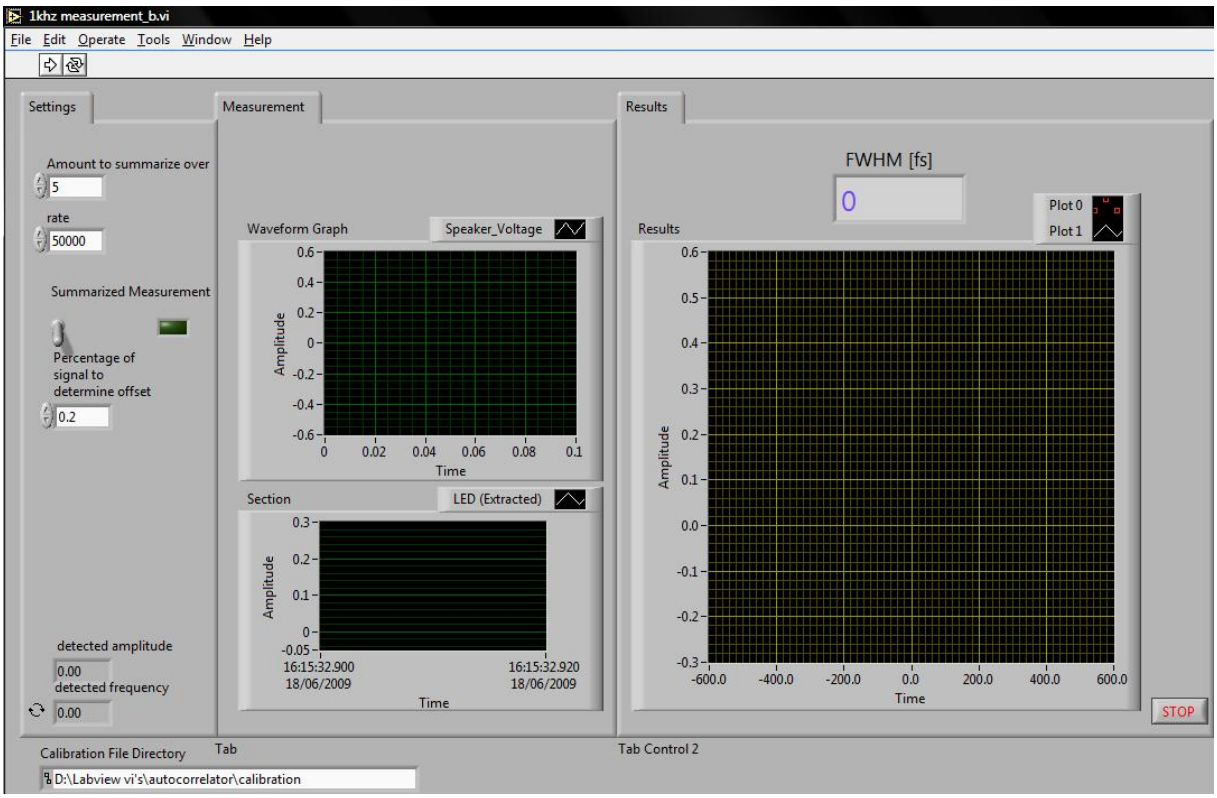
- **Split section 1 way:** Do a peak detection. Split the speaker-signal at 0V (and save the location in a variable called zero) and split the array with peaks at the same location. Count the number of peaks in each section of the splitted peak-array. Send the splitted speaker-signal, amount of peaks in an array-section, the zero-variable and the array-section to the “Fit voltage to location”subVI.
    - **INPUT:**
        - Data (named: Section) (read in data by DAQ-card)
        - Zero 1 (location of the 0-voltage point) [double]
        - Width (Amount of data points for peak-detection) [double]
    - **OUTPUT:**
        - Output array [2D]
  - **Fit voltage to location**: Take for each peak-number the corresponding speaker voltage and place both in an array.
    - **INPUT**:
        - Locations (locations of peaks) [1D-array of double]
        - # found (number of peaks) [long integer]
        - Zero 1 (location of 0-voltage point) [double]
        - Dynamic Data
    - **OUTPUT:**
        - Array [2D]

  Combine results from both “Fit voltage to location” subVI’s and output it.

- **Combine arrays**: Reshapes the variable with multiple dimensions (due to loading a variable amount of files). It determines the minimum and maximum values of each dimension, and reshapes the array so that all the dimensions have the same size (maximum/minimum). It then determines for each peak-number the average measured value, along with the standard deviation and the variance.
    - **INPUT**:
        - Array [3D-array of double]
    - **OUTPUT:**
        - Appended array 2 [2D-array of double]

Final Program

The final program is the program to be used to determine the duration of the laser pulses. It works by collecting the data points that are on the envelope of the pulse, through several measurements, and combining these. It then fits a Gaussian function through the found data, and calculates the full width at half maximum (FWHM). The input for the speaker voltage is channel ai3, and the for the LED-voltage ai1.



1khz.exe / 1khz measurement_b.vi

Amount to summarize over: When running in "summarized measurement mode" the amount of measurements to be done and averaged before showing the output.
Rate: Rate at which the DAQ-card is run.
Summarized measurement: Option to enable or disable the use of a variable running average. If enabled it will store the results of n measurements before calculating the FWHM.
Percentage of signal to determine offset: Amount of datapoints of the measurement that should be used to determine the vertical offset.
Calibration file Directory: Directory where the calibration files can be found.
Results: Shows the envelope of the measured pulse on a fs-scale.
Waveform Graph: Shows the measured speaker and LED voltage.
Section: Shows the section that is used to determine the vertical offset.
FWHM: Measured full width at half maximum.
STOP: self-explanatory.

Run the program by clicking on LABVIEW's "run" button. The program will then ask for the calibration file that is to be used. After selecting the calibration file the program will measure the FWHM of the incoming laser pulse, and display this pulse in the results graph. The FWHM will be calculated and shown. This can be done in two different ways. The first way is to have a continues running average, and the other is by having a running average where the amount of measurements used in calculating the average is limited. The latter option is enabled by flipping the "Summarized measurement" toggle. The LED next to the toggle will "burn" if the summarized measurement function is enabled.

Software's main subVI's:

- **Voltage to femtoseconds:** Take the calibration data. Then look up each voltage value and find the corresponding displacement. Convert that displacement to femtoseconds and place it in an array.
    - **INPUT:**
        - Array (calibration file) [2D of double]
        - Dynamic data (voltage channel)
    - **OUTPUT:**
        - Dynamic data type
- **Increasedvalues:** For each value in a 2-dim array, allow it only to be placed in a new array if it's larger than the previous value. Also place the corresponding value of the other dimension in the array
    - **INPUT:**
        - Dynamic data (output previous subVI)
        - Dynamic data 2 (LED-channel)
    - **OUTPUT:**
        - Graph 1 (LED-signal) [1D-array of double)
        - Array 1 (used voltage-points) [1D-array of double]
        - Array 2 (used led-points) [1D-array of double]
        - Size (size of LED-array) [long integer]
- **Sort 2D array:** Read an input 2D array. Combine each elements for each dimension into a cluster. Sort the clusters and recreate the 2D array, but now sorted.
    - **INPUT:**
        - Array 2 (combined data) [2D-array of double]
        - Disabled index [long integer]
    - **OUTPUT:**
        - Array (sorted data) [2D-array of double]

- **Only max values:** Take the signal and the inverted signal and only allow values of which the previous value was lower. Combine them and place them in an array.
    - **INPUT:**
        - Array 2 (sorted data) [2D-array of double]
    - **OUTPUT:**
        - Array (only larger values) [2D-array of double]
  - **Increasedvalues**